Multi-agent programming

Virtual museum

2012

Mehran Nasseri & Mahboobeh Abdal

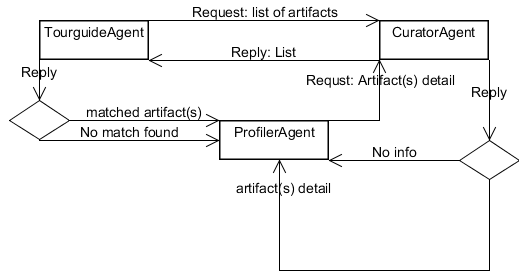
1. **Introduction**

The program includes 3 agents, namely Profiler agent, Curator agent, and Tour guide agent. They interact with each other by implementing several complex and simple behavior.

The Profiler agent asks Tour guide agent for a virtual tour according to the age and preferred art style defined by the user. Each tour comprises name of artifacts and their location (museum / art gallery) that match user preferences. The Profiler agent, then, starts to retrieve the description of each preferred artifact by sending message(s) to each corresponding museum or art gallery.

In addition, the profiler agent repeatedly updates its list of artifacts through continuous communication with curators’ agents.

The interactions among the agent are illustrated in figure1.



**Figure 1:** interaction between agents

1. **Behaviors**
   1. **Profiler Agent**

It gets the age and preferred style of the user as argument before startup. It implements SubscriptionInitiator and overrides its handleInform method to be notified about new available Profiler agent, so it can send a request for virtual tour immediately.

Activities regarding communication with other two agents organized by utilizing SequentialBehavior as follow.

* + 1. **WakerBehaviour:** waits 10 seconds before sending virtual tour request to the Tour guide agent. This is to make sure the new tour guide agent retrieved list of artifacts from curator and has something to offer.
    2. **Generic behavior:** receives reply message from tour guide agent, which either contains list of artifacts match user’s preferences or “no match found” message.
    3. **OneShotBihaviour**: if Tour guide agent replied by proposing a list of artifact, Profiler will start to sends request to curator(s) who owns the artifact(s) in order to retrieve full description of each of them.
    4. **Generic behavior:** receives reply message from curator agent, which either contains description of an artifact or “no match found” message.
  1. **Tour guide agent**

Activities regarding communication with other two agents organized by utilizing ParallelBehavior, since activities are independent and communication with each of them must not interrupt communication with other one.

The child behaviors of the implemented PrallelBehaviour include:

* + 1. **CyclicallBehavior**: handles request received from profiler agent regarding description of a particular artifact.
    2. **TickerBehavior**: searches for registered curator agents in yellow page every minute and sends request to retrieve a list of artifacts. This list, then, will be used to provide a virtual tour for a profiler agent.
    3. **MsgReceiver**: receives list of artifacts from curators. It receives message according to the given message template and timeout. When the timeout expires before any message arrives from curator, the behaviour terminates.
  1. **Curator**

This agent registers its service on yellow page. Curator agent must interact with two other agents in a way that it provides list of artifacts for TourGuide agent and description of specific artifacts for Profiler agent. In order to do this, Curator agent extends ParallelBehaviour to communicate with TourGuide and 'Profiler' agents.

* + 1. **CyclicBehaviuor**: is utilized to reply to Profiler agent. First, the curator agent prepares the message template and set the conversationID as "Artifact-Info" to get message from 'Profiler'. When it receives the request message from 'Profiler', the message will be processed and it tries to find the description of required artifact(s).
    2. **CyclicBehaviuor**: handles the reply process from curator to TourGuide. The message template conversationID is set as "Artifact-list" to receive the message from TourGuide agent. When it receives the message according to the given message template from TourGuide, the request will be processed to reply with the list of artifacts.